AD-A058 791 PURDUE UNIV LAFAYETTE IND SCHOOL OF ELECTRICAL ENGI--ETC F/G 9/4
A PATTERN DEFORMATIONAL MODEL AND BAYES ERROR-CORRECTING RECOGN--ETC(U)
MAY 78 W H TSAI, K S FU AFOSR-74-2661
UNCLASSIFIED TR-EE78-26 AFOSR-TR-78-1297 NL

1 OF 1
AD
A058791

END
DATE
FILMED
11-78
DDC

LEVEL II

②

A PATTERN DEFORMATIONAL MODEL AND BAYES ERROR-CORRECTING RECOGNITION SYSTEM

W. H. Tsai
K. S. Fu

School of Electrical Engineering
Purdue University
West Lafayette, Indiana

TR-EE 78-26
May 1978

This work was partially supported by AFOSR Grant 74-2661 and DARPA MDA Grant 903-77-G-1.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFOSR-TR- 78-1297	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A PATTERN DEFORMATIONAL MODEL AND BAYES ERROR-CORRECTING RECOGNITION SYSTEM		5. TYPE OF REPORT & PERIOD COVERED Interim
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) W.H. Tsai and K.S. Fu		8. CONTRACT OR GRANT NUMBER(s) AFOSR 74-2661
9. PERFORMING ORGANIZATION NAME AND ADDRESS Purdue University Department of Electrical Engineering Lafayette, Indiana 47907		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61102F 2304/A2
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Office of Scientific Research/NM Bolling AFB, Washington, DC 20332		12. REPORT DATE May 1978
		13. NUMBER OF PAGES 43
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Various types of pattern deformations are investigated from the syntactic point of view and categorized into two major types: local deformations and structural deformations. Random noise, distortion variations, and substitutions, of pattern primitives belong to the former; syntactic errors due to pattern structural changes, such as primitive deletions and insertions, belong to the latter. Every observed pattern can be regarded as transformed from a pure pattern through these two types of deformations. An error-correcting parsing scheme for local deformations optimum in the Bayes sense

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

20. Abstract

is proposed. A corresponding recognition rule is then described, which can be regarded as a hybrid classifier because it has utilized advantages of both syntactic and statistical approaches to pattern recognition. When this scheme is applied to string and tree languages without structural deformations, it is shown that various known structure-preserved error-correcting parsing schemes could be considered as special cases of this general scheme. Two structure-preserved error-correcting parsers, one for string languages, the other tree languages, are also presented. Finally, further researches concerning error-correcting parsings for structural deformations and a complete error-correcting systems for both kinds of deformations are suggested.

UNCLASSIFIED

9 Interim rept.,

6

A PATTERN DEFORMATIONAL MODEL AND BAYES ERROR-CORRECTING RECOGNITION SYSTEM

10 W. H. Tsai [redacted] K. S. Fu

School of Electrical Engineering

Purdue University

West Lafayette, Indiana 47907

14 TR-EE 78-26

11 May [redacted]78

12 43p.

18 AFOSR

19 TR-78-1297

16 2304

17 A2

15

This work was partially supported by AFOSR-[redacted] 74-2661 [redacted]RPA
MDA Grant 903-77-G-1,

MDA903-77-G-1

292 000

78 09 13 121

Gu

ABSTRACT

Various types of pattern deformations are investigated from the syntactic point of view and categorized into two major types: local deformations and structural deformations. Random noise, distortion variations, and substitutions, of pattern primitives belong to the former; syntactic errors due to pattern structural changes, such as primitive deletions and insertions, belong to the latter. Every observed pattern can be regarded as transformed from a pure pattern through these two types of deformations. An error-correcting parsing scheme for local deformations optimum in the Bayes sense is proposed. A corresponding recognition rule is then described, which can be regarded as a hybrid classifier because it has utilized advantages of both syntactic and statistical approaches to pattern recognition. When this scheme is applied to string and tree languages without structural deformations, it is shown that various known structure-preserved error-correcting parsing schemes could be considered as special cases of this general scheme. Two structure-preserved error-correcting parsers, one for string languages, the other for tree languages, are also presented. Finally, further researches concerning error-correcting parsings for structural deformations and a complete error-correcting systems for both kinds of deformations are suggested.

1. Introduction

To recognize noisy or deformed patterns using the syntactic pattern recognition approach, error-correcting parsing and classification techniques using various decision criteria have been proposed [1-5,20]. Errors induced on the primitives of noisy or deformed patterns usually are classified into three types: substitutions, deletions, and insertions. If only substitution errors are considered, the error-correcting parser is said to be structure-preserved. After an input pattern is parsed by a certain pattern grammar, a quantitative measure, either deterministic or probabilistic, is output by the parser to indicate a measure of possibility that the input pattern is generated by the grammar. The decision criterion is then used to classify the input pattern as belonging to the pattern class with an extreme quantitative measure, either minimum or maximum, depending on how the measure is defined. Two most widely used decision criteria are minimum-distance and maximum-likelihood criteria, though others have also been proposed [2,5].

Influenced by the linguistic types of representation which only adopts symbolic notations as terminals, most of the existing error-correcting parsing methods [1-4,20] use discrete symbols to represent structural pattern primitives. However, it happens quite often that a primitive also contains continuous semantic or numerical information useful for pattern discrimination purpose [5,6,7]. For such cases, obviously, these parsing methods are not appropriate, because they can not utilize continuous semantic or numerical information.

To take care of both structural and numerical information simultaneously, a deformational model for pattern primitives is introduced in this report. Based on this model, error-correcting parsing and classification techniques using the Bayse decision rule are then proposed. Various known error-correcting parsing schemes and classification rules are compared with the proposed techniques. A complete illustrative example is given to show the applicability of the proposed model and techniques.

2. A Deformational Model

In this section, we give a formal description of basic concepts for images, patterns, subpatterns, and primitives, which we will call structural entities, used in syntactic pattern recognition from a broader point of view, and based on these concepts, we propose a deformational model which will serve as a basis later for developing a Bayes error-correcting recognition system. Essentially, these concepts are described as general as possible so that they can be applied to a variety of pattern languages, and in such a way that discrimination between syntactic and semantic informations available from the structural entities is emphasized. In particular, examples are given for string and tree languages for illustrative purpose.

2.1 Basic Concepts

An observed image usually can be considered as deformed from a pure image. For example, a smooth shape in a picture may become noisy after it is digitized. Here the original shape is the pure image and its noisy version is the observed image. When similar pure images are clustered as a pure pattern class, there corresponds a set of observed images each of which we will call as an observed pattern. In practical applications, grammars are often inferred, either from pure or from observed patterns, to recognize observed images. In some simple cases, the deformations, such as noises, existing in observed patterns can be eliminated by intensive preprocessing such as thresholding. But in general, they can not be eliminated entirely. This is why error-correcting parsings are necessary.

Before a class of patterns can be described by a pattern grammar, each pattern is decomposed into smaller and simpler structural units called primitives. Primitives should be chosen properly so that the resulting descriptions of the patterns using grammars can be simple [7]. We call the description of a pattern using some fixed primitives as a structural representation, which is, for string languages, a string (representation) consisting of symbols each of which corresponds to a primitive, and is, for tree languages, a tree (representation) with each of its nodes corresponding to a primitive. Of course, pure primitives, pure patterns, and pure structural representations also have their corresponding observed primitives, observed patterns, and observed structural representations, respectively.

2.2 Primitives

A detailed study of various kinds of primitives used for pattern descriptions [7-9] reveals that each primitive may contain two kinds of information, namely, the syntactic information and the semantic information. The syntactic information gives a structural description of the primitive, and the semantic information provides the meaning or numerical description of the primitive. To be more specific, two examples are given in the following for illustrative purpose.

I. Primitives for string languages --- A primitive for string languages usually is simply a symbol. Different symbols are used to represent different primitives, such as an arc, a straight line segment, an angle, etc., for describing shape boundaries. But it happens quite often that we need more information involving numerical measurements to describe a primitive more accurately. For example, we may want to discriminate two arc primitives by their lengths and curvatures. Then, the syntactic information contained in these two primitives is the arc structure, and the semantic information is their respective lengths and curvatures. You and Fu [9] used two kinds of primitives - curve segment primitives and angle primitives - to describe shapes. The first one is a curve segment with 4 numerical features to describe its direction, length, curvature, and symmetry. The second one is an angle with one feature to describe the angle amplitude. These two kinds of primitive serve as a very good example for illustrating the above concept of primitive information.

II. Primitives for tree languages --- Similarly, a primitive for tree languages may have any kind of primitive structure and various kinds of numerical measurements on the primitive. For example, Lu and Fu [10] used a pixel with it gray value as a primitive to set up a tree model. Then the primitive structure is a pixel and the semantic information is its gray value.

Now we are ready to give a formal description of a primitive. We consider a primitive a, either pure or observed, as a 2-tuple

$$a = (s,x)$$

where

s is a <u>syntactic symbol</u> densting the primitive structure of a, and

$x = [x_1,x_2,\ldots,x_m]$ is an m-dimensional <u>semantic vector</u> with each x_i ($i = 1,2,\ldots,m$) denoting a numerical measurement or a logical predicate, and $m \geq 0$. When $m = 0$, or no semantic information is available, set $x = \phi$ (empty vector).

A similar idea was also proposed by Shaw [21] and described in Fu [7].

Two remarks are in order.

I. Influenced by the linguistic represenations, the primitives used in syntactic pattern recognition tend to be restricted to symbolic notations which essentially only give syntactic information. Even when a continuous type of numerical information, such as random noise, is included in the primitives, it is often thresholded into discrete numericals which then are denoted by a finite number of primitive symbols. Such an approach not only decreases the discrimination accuracy due to the numerical thresholding but also increases the number of grammar rules due to the increase of the number of primitives (i.e. terminals). With a primitive described as above, such weaknesses could be eliminated as will be seen later.

II. Since a primitive contains two parts of information, we obtain a great deal of flexibility in selecting primitives. This point is also emphasized in [6]. Any structural unit can be selected as a primitive, and if more properties are needed to specify the primitive, numerical or semantic information can be invoked. Furthermore, with semantic information separated from syntactic information in a primitive, a very systematic deformational model can be developed for optimum error-correcting parsing schemes which will be described in the following sections.

2.3 Pattern Structures

To transform a pattern into a structural representation using primitives as constructing units, we need a fixed constructing rule which we will call a pattern structure. For example, to convert a shape into a string representation with arcs, line segments, angles as primitives, we have to know the starting primitive and the direction the shape boundary should be traced. So a string structure is needed. Similarly, a tree structure is needed to convert the set of primitives of a given pattern into a tree representation (for example, see [5,19]). So a structural representation of a pattern can be considered as the arrangement of primitives according to a fixed pattern structure. Usually, in practical applications, the number of pattern structures used by a pattern language is finite and not too large. In some cases, there is even only one such structure used for all structural representations [5,10]. For string languages, strings with different lengths are of different string structures, and for tree languages, trees with different number of nodes or different connecting branches are also of different tree structures. But the number of primitives existing in a structural representation is not the only discriminant factor of pattern structures. In some cases, different implicit relations implied by the concatenations in a string or by the branches in a tree also define different pattern structues, although such relations may be represented explicitly by terminals by some pattern languages such as PDL and PLEX languages [21,22].

Now we can say that a pattern class consists of a set of patterns each of which in turn can be transformed into a structural representation using a set of prespecified primitives (and relations) according to one of some fixed pattern structures for this pattern class. These structural representations can then be used to infer a pattern grammar to characterize this pattern class. So each terminal used in the grammar is just a primitive which can be described by a 2-tuple consisting of a syntactic symbol and a semantic vector as defined in Section 2.2.

2.4 The Deformational Model

From previous discussions, it is clear that a pattern or its structural representation ω can be fully characterized by a 2-tuple $\omega = (S,A)$ where $A = \{a_i | i = 1,2,...,n\}$ is a set of primitives used in ω and S denotes the pattern structure of ω together with implicitly assumed relations among the primitives. For discussion convenience in the following sections, we assume that the subscripts for a_i are numbered according to some fixed order which is determined by the pattern structure S; when S is fixed, then this ordering is also fixed.

Given the structural representation $\omega = (S,A)$ of a certain pure pattern with pattern structure S and primitive set

$$A = \{a_i | a_i = (s_i, x_i),\ x_i = (x_{i1}, x_{i2}, \ldots, x_{iN_i}),\ N_i \geq 0,\quad i=1,2,\ldots,n\},$$

the structural representation of its corresponding observed pattern $\omega' = (S', A')$, with pattern structure S' and primitive set

$$A' = \{a'_i | a'_i = (s'_i, x'_i),\ x'_i = (x'_{i1}, x'_{i2}, \ldots, x'_{iN'_i}),\ N'_i \geq 0,\ i=1,2,\ldots,n\},$$

can be considered as being transformed from ω through a series of deformations. Our deformational model categorizes all possible deformations int two major types: structural deformations and local deformations.

I. Local deformations --- If $S = S'$, but for some i, $i = 1,2,\ldots,n$, $a_i \neq a'_i$, then we say ω' is deformed locally from ω. In another word, a local deformation induced on a pure pattern preserves the entire pattern structure but deforms some primitives locally. So a local deformation is also called a structure-preserved deformation. With respect to strings, this simply means a length-preserved deformation.

II. Structural deformations --- If $S \neq S'$, then we say that ω' is deformed structurally from ω. Various types of structural deformations, such as insertions, deletions, transpositions, and permutations [11,2,12], have been defined according to various kinds of structural difference between S and S'.

In this report, we deal only with local deformations, leaving structural deformations for further investigations.

2.5 Local Deformations

A deformation induced on at least one primitive of a given pure pattern is called a local deformation. Let $a_i = (s_i, x_i)$ be the pure primitive deformed, where

$$x_i = (x_{i1}, x_{i2}, \ldots, x_{iN_i}),$$

and $c_i = (t_i, z_i)$ be one of its observed versions, where

$$z_i = (z_{i1}, z_{i2}, \ldots, z_{iN'_i}).$$

At least two types of local deformations can be identified as following:

I. Syntactic local deformation --- This is the case when $t_i \neq s_i$. In another word, when the primitive structure is changed to another one, a *syntactic local deformation* is induced, which usually is called a *substitution error*.

II. Semantic local deformation --- When the local deformation on a_i does not change the primitive structure but only corrupts the semantic information, i.e. when $t_i = s_i$ but $z_i \neq x_i$, then it is called a *semantic local deformation*. If every primitive used by a pattern has an identical primitive structure, then every local deformation is semantic.

In general, we can consider a local deformation as a two-step transformation from $a_i = (s_i, x_i)$ to $c_i = (t_i, z_i)$ by the following way:

$$\begin{array}{ccccc} (s_i,x_i) & \xrightarrow{\text{synt.loc.def}} & (t_i,y_i) & \xrightarrow{\text{sem.loc.def}} & (t_i,z_i) \\ \uparrow & & \uparrow & & \uparrow \\ \text{pure primit.}a_i & & \text{semi-pure primit.}b_i & & \text{observed primit.}c_i \end{array}$$

where $b_i = (t_i, y_i)$, called a *semi-pure primitive*, is created to denote one of the syntactically local-deformed versions of (s_i, x_i) with y_i being a *representation semantic vector* for t_i, which is only created for explanatory convenience and does not have much practical use later in our derivation of parsing procedures.[†] When $t_i = s_i$, then $y_i = x_i$, and only semantic local deformations happen in the two-step transformation.

Let $A = \{a_i | a_i = (s_i,x_i),\ i = 1,2,\ldots,n\}$ denote all the pure primitives used in a pure pattern. Though each a_i can be deformed syntactically into a set of semi-pure primitives $D_{a_i} = \{b_{ij} | b_{ij} = (t_{ij},y_{ij}),\ j = 1,2,\ldots,k_i\}$, each deformation $a_i \to b_{ij}$ may occur with a different probability. So there exists a conditional probability function p defined on D_{a_i} for each a_i such that $p(b_{ij}|a_i) = p(t_{ij}|s_i)$ is the probability for s_i to be deformed into t_{ij}, $j = 1,2,\ldots,k_i$. Similarly, since each b_{ij} can be deformed semantically into a set of observed primitives $D_{b_{ij}} = \{c_{ijk} | c_{ijk} = (t_{ij},z_{ijk}),\ z_{ijk} \in R_{ij}\}$, where R_{ij} is a range for z_{ijk}, which may consist of a finite number of discrete elements or of an infinitive number of continuous elements, we can define a conditional probability or density function q on $D_{b_{ij}}$ such that $q(z_{ijk}|b_{ij},a_i) = q(z_{ijk}|t_{ij},s_i)$ is the probability or density for $b_{ij} = (t_{ij},y_{ij})$ to be deformed into $c_{ijk} = (t_{ij},z_{ijk})$. Therefore, from a probabilistic point of view, a local deformation from $a_i = (s_i,x_i)$ to $c_{ijk} = (t_{ij},z_{ijk})$ now can be interpreted as following:

$$a = (s_i,x_i) \xrightarrow[\text{synt.loc.def.}]{p(t_{ij}|s_i)} b_{ij} = (t_{ij},y_{ij}) \xrightarrow[\text{sem.loc.def.}]{q(z_{ijk}|t_{ij},s_i)} c_{ijk} = (t_{ij},z_{ijk}),$$

where $p(\cdot|s_i)$ is the conditional probability function given a_i (or s_i) defined on D_{a_i}, and $q(\cdot|t_{ij},s_i)$ is the conditional probability or density function given a_i and b_{ij} (or s_i,t_{ij}) defined on $D_{b_{ij}}$. We also assume that $a_i \in D_{a_i}$, and $b_{ij} \in D_{b_{ij}}$.

To be more specific, we give two examples for the semantic local deformations, assuming no syntactic local deformation is involved --- that is,

†Sometimes for normally distributed z_i, y_i can be conveniently chosen to be the mean value of z_i.

$$a_i = (s_i, x_i) \xrightarrow[\text{sem.loc.def.}]{q(z_{ij}|s_i)} c_{ij} = (s_i, z_{ij}) .$$

I. Random noise --- This is the case when the semantic vector x_i in a pure primitive $a_i = (s_i, x_i)'$ is subject to random noise corruption. So the deformed or noisy version of x_i, denoted as z_{ij} above, is actually a random vector $\underline{z}_{ij}$ with continuous density function $q(\cdot|s_i)$. If the noise associated with $\underline{z}_{ij}$ is normally distributed with zero mean, then x_i in fact is just the mean vector of $\underline{z}_{ij}$, or $x_i = E\{\underline{z}_{ij}\}$.

II. Distortion variations --- In some cases, x_i may be deformed into only a finite number of observed versions z_{ij}. Then $q(\cdot|s_i)$ above is just a discrete probability function defined on all possible z_{ij}.

Back to our discussion of two-step local deformations, given a pure primitive $a_i = (s_i, x_i)$, the probability that it is deformed locally into an observed primitive $c_{ijk} = (t_{ij}, z_{ijk})$ now can be computed as

$$\gamma(c_{ijk}|a_i) = \lim_{\Delta z_{ijk} \to 0} p(t_{ij}|s_i) \cdot q(z_{ijk}|t_{ij}, s_i) \cdot \Delta z_{ijk}$$

if $q(\cdot|t_{ij}, s_i)$ is a continuous density function, or simply

$$\gamma(c_{ijk}|a_i) = p(t_{ij}|s_i) q(z_{ijk}|t_{ij}, s_i)$$

if $q(\cdot|t_{ij}, s_i)$ is a discrete probability function. And given a pure pattern $\omega = (S, A)$ with $A = \{a_i | a_i = (s_i, x_i),\ i = 1,2,\ldots,n\}$, the probability that ω is deformed locally into a structure-preserved observed pattern $\omega' = (S, C)$ with $C = \{c_i | c_i = (t_i, z_i),\ a_i \xrightarrow{\text{loc.def}} c_i,\ i = 1,2,\ldots,n\}$ is

$$P(\omega'|\omega) = \prod_{i=1}^{n} \gamma(c_i|a_i)$$

$$= \prod_{i=1}^{n} \lim_{\Delta z_i \to 0} p(t_i|s_i)q(z_i|t_i,s_i) \cdot \Delta z_i \ ,$$

when $q(\cdot|t_i)$ is continuous, or,

$$P(\omega'|\omega) = \prod_{i=1}^{n} p(t_i|s_i)q(z_i|t_i,s_i) \ ,$$

when $q(\cdot|t_i)$ is discrete, if each a_i is deformed <u>independently</u> into c_i, $i = 1,2,\ldots,n$. such independence assumption for local deformations of primitives was also considered by Grenander [13], Kovalevsky [14], and Fung and Fu [3].

3. Bayes Structure-Preserved Error-Correcting Parsers

In this section, we derive structure-preserved error-correct parsers (SPECP) optimum in the Bayes sense for locally deformed patterns. Given a pattern class consisting of various pure patterns which can be generated by a pattern grammar, we can, from statistical point of view, consider each pure pattern together with all its possible locally deformed versions as a distinct subclass of the given pattern class. Then the SPECP to be derived, which we will call Bayes SPECP, are optimum in the sense that they are, in addition to possessing syntactic parsing capability, just Bayes subclass classifiers which assign each given observed pattern, according to Bayes decision rule, to a subclass whose pure pattern has a maximum probability to be deformed into the given observed pattern.

3.1 Statistical Considerations

Given an observed pattern $\omega = (S,A)$ with $A = \{a_i | a_i = (s_i,x_i),\ x_i = (x_{i1},x_{i2},\ldots,x_{iL_i}),\ i = 1,2,\ldots,n\}$ of a certain pure pattern class C which consists, for simplicity, of only two pure patterns $\omega_1 = (S,B_1)$ and $\omega_2 = (S,B_2)$ with $B_1 = \{b_i^1 | b_i^1 = (t_i^1,y_i^1),\ y_i^1 = (y_{i1}^1,y_{i2}^1,\ldots,y_{iM_i^1}^1),\ i = 1,2,\ldots,n\}$ and $B_2 = \{b_i^2 | b_i^2 = (t_i^2,y_i^2),\ y_i^2 = (y_{i1}^2,y_{i2}^2,\ldots,y_{iM_i^2}^2),\ i = 1,2\ldots,n\}$, we want to assign ω to one of the two pure pattern subclasses ω_1 and ω_2 according to the statistical hypothesis testing theory. Using the Bayes decision rule, we get, according to the analysis for the deformational model in the last section under the independence assumption for local deformations,

$$\frac{P(\omega_1|\omega)}{P(\omega_2|\omega)} \lessgtr 1 \qquad \text{decide} \qquad \omega \to \begin{cases} \omega_1 \\ \omega_2 \end{cases},$$

or

$$\frac{P(\omega|\omega_1)P(\omega_1)}{P(\omega|\omega_2)P(\omega_2)} = \left[\prod_{i=1}^{n} \frac{\gamma(a_i|b_i^1)}{\gamma(a_i|b_i^2)}\right] \cdot \frac{P(\omega_1)}{P(\omega_2)}$$

$$= \left[\prod_{i=1}^{n} \frac{p(s_i|t_i^1)q(x_i|s_i,t_i^1)}{p(s_i|t_i^2)q(x_i|s_i,t_i^2)}\right] \cdot \frac{P(\omega_1)}{P(\omega_2)} \gtrless 1 \text{ decide } \omega \to \begin{cases}\omega_1 \\ \omega_2\end{cases},$$

or taking logarithms,

$$\sum_{i=1}^{n} [\ln p(s_i|t_i^1) + \ln q(x_i|s_i,t_i^1)] + \ln P(\omega_1)$$

$$\gtrless \sum_{i=1}^{n} [\ln p(s_i|t_i^2) + \ln q(x_i|s_i,t_i^2)] + \ln P(\omega_2)$$

$$\text{decide } \omega \to \begin{cases}\omega_1 \\ \omega_2\end{cases},$$

where $P(\omega_1|\omega)$, $P(\omega_1|\omega)$, $P(\omega_1)$, $P(\omega_2)$ are posteriori and a priori probabilities for pure pattern subclass ω_1 and ω_2, and $p(\cdot|t_i^j)$, $q(\cdot|s_i,t_i^j)$, $j = 1,2$, are as defined in the last section. When the pure pattern class C consists of more than two patterns, the above decision rule can be extended as following. Let λ_i be such that

$$-\ln \lambda_j = -\sum_{i=1}^{n} [\ln p(s_i|t_i^j) + \ln q(x_i|s_i,t_i^j)] - \ln P(\omega_j),$$

$j = 1,2,\ldots,M$, with M, either finite of infinite, being the total number of pure patterns belonging to C, then decide $\omega \to \omega_k$ if k is such that

$$-\ln \lambda_k = \min_{j=1,2,\ldots,M} (-\ln \lambda_j) .$$

We call the term $-\ln \lambda_j$ the <u>Bayes distance</u> $B(\omega,\omega_j)$ from ω to ω_j, and the term $-\ln \lambda_k$ the <u>minimum Bayes distance</u> $B(\omega,C)$ from ω to pure pattern class C.

With Bayes distances defined as above, the <u>Bayes SPECP</u>, constructed from the pattern grammar G_c for a given pure pattern class C, is used to search for a given input observed pattern ω a pure pattern ω_k accepted by G_c with a minimum Bayes distance $B(\omega,\omega_k) = B(\omega,C)$ during the error-correcting parsing. So our problem now is reduced to how to compute the Bayes distances $-\ln \lambda_j$ <u>during the parsing procedure</u>. Since the parsing is done on each primitive at least once, there is no problem in obtaining the first term $\sum_{i=1}^{n} [p(s_i|t_i^j) + \ln q(x_i|s_i,t_i^j)]$ in $-\ln \lambda_j$, as will be seen later. But how to get the a priori probability $P(\omega_j)$ for the pure pattern ω_j <u>during the parsing procedure</u> is on the contrary not so obvious. The solution is to use a stochastic grammar for the pattern class C.

3.2 <u>Stochastic Grammars for Computing Pattern Probabilities</u>

Stochastic grammars have been introduced to take care of noisy patterns and also to specify the probability of occurrence for each pattern accepted by the pattern grammars [7]. The latter property is exactly what we want for computing pattern probabilities $P(\omega_j)$.

To be more specific, a stochastic grammar is a grammar each of whose production rules is associated with an occurrence probability. When a stochastic pattern grammar is used to generate the structural representation of a given pattern, a pattern occurrence probability is also generated simultaneously, which is the product of all probabilities associated with the production rules used in deriving the structural representation. For details, see Fu [7]. And for inference of production rule probabilities, see Lee and Fu [15]. Here we only give the basic notations and definitions of stochastic context-free grammars and stochastic tree grammars [7].

Definition 1. A stochastic context-free (string) grammar is a 4-tuples $G_s = (V_N, V_T, P_s, S)$, where

V_N is a finite set of non-terminals,

V_T is a finite set of terminals,

S is a start symbol,

P_s is a finite set of stochastic production rules, each of which is of the form

$$A_i \overset{p_{ij}}{\rightarrow} \alpha_{ij},\ j = 1,2,\ldots,n_i\ ,\ i = 1,2,\ldots,\ell\ ,$$

where $A_i \in V_N$, $\alpha_{ij} \in (V_T \cup V_N)^*$, n_i is the number of distinct production rules with A_i at left-hand side, ℓ is the number of nonterminals, and p_{ij} is the probability associated with this production. Furthermore,

$$0 < p_{ij} \leq 1 \quad \text{and} \quad \sum_{j=1}^{n_i} p_{ij} = 1\ .$$

Definition 2. A stochastic context-free (string) grammar G_s is in Chomsky normal form if each of its production rule is of the formm

$$A \overset{p}{\rightarrow} \overset{BC}{\cancel{aB}} \quad \text{or} \quad A \overset{p}{\rightarrow} a$$

where $A, B, C \in V_N$, $a \in V_T$.

Definition 3. A stochastic tree grammar over $\langle V_T, r\rangle$ in its expansion form is a 4-tuple $G_t = (V_N \cup V_T, r, P, S)$, where

V_N, V_T, S are the same as defined in Definition 1,

$r: V_T \to N$, the set of nonnegative integers, is a rank function denoting the number of direct descendants of a node with a symbol in V_T as its label, and

P is a set of stochastic production rules, each of which is in the form

$$X_i \xrightarrow{p_{ij}} \begin{array}{c} x \\ \diagup \quad \diagdown \\ X_{i1} \cdots X_{ir(x)} \end{array} \qquad \text{or} \qquad X_i \xrightarrow{p_{ij}} x$$

where $x \in V_T$, $X_i, X_{i1}, \ldots, X_{ir(x)} \in V_N$, $1 \le j \le n_i$, $1 \le i \le \ell$, n_i, ℓ, p_{ij} are the same as defined in Definition 1, and

$$0 < p_{ij} \le 1 \quad \text{and} \quad \sum_{j=1}^{n_i} p_{ij} = 1 .$$

3.3 Bayes SPECP for String Languages

We describe in the following a Bayes SPECP for context-free string languages. Given a stochastic context-free string grammar $G_s = (V_N, V_T, P_s, S)$ for a pure pattern class, assume that the terminal set $V_T = \{a_i | a_i = (t_i, w_i),\ i = 1,2,\ldots,\ell\}$ contains all possible pure primitives used by the pure patterns. For each a_i, $i = 1,2,\ldots,\ell$, let $p(\cdot|a_i) = p(\cdot|t_i)$ be the conditional probability function defined on $D_{a_i} = \{b_{ij} | b_{ij} = (u_{ij}, y_{ij}),\ a_i \xrightarrow[\text{syn.loc.def}]{} b_{ij},\ j = 1,2,\ldots,k_i\}$, and $q(\cdot|a_i, b_{ij}) = q(\cdot|t_i, u_{ij})$ be the conditional probability or density function defined on $D_{b_{ij}} = \{c_{ijk} | c_{ijk} = (u_{ij}, z_{ijk}),\ b_{ij} \xrightarrow[\text{sem.loc.def.}]{} c_{ijk},\ c_{ijk} \in R_{ij}\}$. Let

$$V_T' = \bigcup_{i=1}^{\ell} [\bigcup_{j=1}^{k_i} D_{b_{ij}}]$$

denote all possible deformed primitives, and note that $V_T \subset V_T'$. The algorithm for the Bayes SPECP is a modification of the Cocke-Yonger-Kasami parsing scheme [16], which essentially tries to construct a parse table T for an input observed string representation y, and then parses through the table to obtain a pure string representation x with a minimum Bayes distance $B(y,x)$. The table T consists of entries t_{ij}, $1 \leq i \leq n$, $1 \leq j \leq n-i+1$, where n is the length of string y. Each t_{ij} is a set of triplets (A,d,k), where $A \in V_N$ is an intermediate nonterminal used in deriving x, $d \in (0,\infty)$ is part of the Bayes distance, and k specifies the product rule used with A at the left-hand side.

<u>Algorithm 1</u>. Bayes Structure-Preserved Error-Correcting Parser for String Languages

<u>Input</u>: A stochastic context-free string grammar $G_s = (V_N, V_T, P_s, S)$ in Chomsky normal form without ϵ-productions, and an observed string representation $y \in V_T'^{*}$, $y = c_1c_2...c_n$, $c_i = (s_i, x_i)$, $i = 1,2,...,n$.

<u>Output</u>: A pure string representation x accepted by G_s with a minimum Bayes distance $B(y,x)$.

<u>Method</u>: Put all production rules into order and let $k: A \xrightarrow{p} \alpha$ denote that $A \xrightarrow{p} \alpha$ is the kth rule in P_s.

<u>Step 1</u>. Construct t_{i1} for each i, $i = 1,2,...,n$. Let $A \in V_N$. For every $k_j: A \xrightarrow{p_j} a_j$ in P_s, $j = 1,2,...,n_A$, where $a_j = (t_j, w_j)$, n_A is the number of production rules each with A on the left-hand side and a terminal on the right-hand side, let

$$d_{ij} = -[\ell n\, p(s_i|t_j) + \ell n\, q(x_i|t_j,s_i) + \ell n\, p_j] ,$$

$i = 1,2,\ldots,n$. Then set

$$t_{i1} = \{(A, d_{i\ell}, k_\ell) | d_{i\ell} = \min_{j=1,2,\ldots,n_A} d_{ij} , A \in V_N\} .$$

Step 2. Construct t_{ij}, $j = 2,\ldots,n$, inductively. Assume that $t_{ij'}$ has been computed for all i, $1 \leq i \leq n$, and for all j', $1 \leq j' < j$. For every $k_j: A \xrightarrow{p_j} B_jC_j$, $j = 1,2,\ldots,n_A'$, where n_A' is the number of production rules with A on the left-hand side and two nonterminals on the right-hand side, if there exists some m, $1 \leq m \leq j$, such that $(B_j,e_{j1},h_{j1}) \in t_{im}$ and $(C_j,e_{j2},h_{j2}) \in t_{i+m,j-m}$, let $e_{ij} = e_{j1} + e_{j2} - \ell n\, p_j$. Then set

$$t_{ij} = \{(A,e_{i\ell},k_\ell)|e_{i\ell} = \min_{j=1,2,\ldots,n_A'} e_{ij} , A \in V_N\}$$

Step 3. Repeat Step 2 until t_{ij} is computed for all $1 \leq i \leq n$ and $1 \leq j \leq n-i+1$.

Step 4. When the entire table T is completed, exam entry t_{1n}. If there exists a triplet (S,d,k) in t_{1n} for some d and k, then set $B(y,x) = e^{-d}$, and the desired pure string representation x can be easily traced out from the parse table T, starting from the kth production rule. If no (S,d,k) exists in t_{1n}, then input observed string representation y is not structure-preserved; set $B(y,x) = 0$.

3.4 Bayes SPECP for Tree Languages

Using the minimum-Bayes-distance criterion again, we propose a Bayes SPECP for tree languages in the following. Given a stochastic tree grammar $G_s = (V_N \cup V_T, r, P_s, S)$ over $\langle V_T, r\rangle$ in its expansive form, let V_T, $p(\cdot|a_i) = p(\cdot|t_i)$, $q(\cdot|a_i,b_{ij}) = q(\cdot|t_i,u_{ij})$, D_{a_i}, $D_{b_{ij}}$, and V_T' be all the same as those defined in Sec. 3.3. The algorithms for the Bayes SPECP follows the concept of tree automata [17], and is a backward procedure for constructing a tree-like transition table T for an input observed tree representation β. Let the tree structure (i.e., the tree domain) of β be denoted as D_β, then corresponding to each node b in D_β is an entry t_b in T, which consists of a set of triplets (A,d,k), where $A \in V_N$ is a candidate state for node b, d is part of the Bayes distance, and k specifies the production rule used with A at the left-hand side.

Algorithm 2. Bayes Structure-Preserved Error-Correcting Parser for Tree Languages

Input: A stochastic tree grammar $G_s = (V_N \cup V_T, r, P_s, S)$ over $\langle V_T, r \rangle$ in its expansive form, and an observed tree representation β with $\beta(b) = (s_b, x_b)$ as its observed primitive at node b, $(s_b, x_b) \in V_T'$.

Output: A pure tree representation α accepted by G_s with a mininum Bayes distance $B(\beta,\alpha)$.

Method: Let $t_{b \cdot i}$ denote the set of triplets corresponding to the ith descendant of node b.

Step 1. For each node b in β such that $r[\beta(b)] = 0$, add to t_b a triplet (A,d,k) with

$$d = -[\ln p(s_b | t_k) + \ln q(x_b | t_k, s_b) + \ln p_k]$$

if

$$A \xrightarrow{p_k} a_k$$

with $a_k = (t_k, w_k)$ is the kth production rule in P_s.

Step 2. For each node b in β such that $r[\beta(b)] = N \neq 0$, add to t_b a triplet (A, d_s, k) with

$$d_s = -[\ln p(s_b|t_k) + \ln q(x_b|t_k, s_b) + \ln p_k]$$

$$+ d_1 + d_2 + \ldots + d_N$$

if

$$A \xrightarrow{p_k} \begin{array}{c} a_k \\ / \quad \backslash \\ A_1 \cdots A_N \end{array}$$

with $a_k = (t_k, w_k)$ is the kth production rule in P_s and $(A_1, d_1, k_1) \in t_{b\cdot 1}$, $(A_2, d_2, k_2) \in t_{b\cdot 2}, \ldots, (A_N, d_N, k_N) \in t_{b\cdot N}$.

Step 3. For any two triplet (B_1, d_1, k_1), (B_2, d_2, k_2) in each t_b, delete the former if $d_1 \geq d_2$, or the latter if $d_1 < d_2$.

Step 4. Repeat Steps 1-3 until all nodes in β have been processed.

Step 5. Exam t_0, the root entry of the transition table T. If $(S, d, k) \in t_0$ for some d and k, then set $B(\beta, \alpha) = e^{-d}$, and the desired pure tree representation α can be easily traced out from T, starting from the kth production rule in P_s. If no (S, d, k) exists in t_0, then the input observed tree representation β is not structure-preserved; set $B(\beta, \alpha) = 0$.

3.5 Comments on Various SPECP and Least-Square-Error Distance Criteria

Fung and Fu [3] have proposed a maximum-likelihood SPECP for string languages, but the grammars used are nonstochastic, so their SPECP is just a suboptimum one under the assumption that all pattern subclasses occur with an equal probability. SPECP using stochastic grammars has been proposed by Fung and Fu [18], Lu and Fu [10,20], and Thompson [2], but from the view point of our deformational model, their SPECP for substitution error only takes care of syntactic local deformations, and so limit their applicability to pattern classification problems where the semantic information, especially when it is continuous, is contained in the pattern primitives for discrimination purpose. Of course, these SPECP still can be used to handle continuous types of semantic information by thresholding them into finite discrete cases, but obviosuly this will decrease the error-correcting capability of the SPECP, as mentioned previously in Sec. 2.2, and as will be shown by an example in Sec. 4.1.

Next, SPECP for string and tree languages using the minimum-distance criterion have also been proposed [1,4]. In addition to being limited to syntactic local deformations, these SPECP are statistically optimum only under very special conditions, although they are convenient and important in practical applications when deformation probabilities or density functions are difficult to infer.

Finally, we propose in the following a new criterion, namely, the <u>least-square-error</u> <u>(LSE)</u> <u>distance</u> criterion for the SPECP, which is a special case of the minimum-Bayes-distance criterion but is useful for semantic local deformations.

It happens sometimes that the observed semantic vector in a primitive is normally distributed, especially when it is computed with random noise. Assuming that no syntactic local deformation involves, we want to derive the Bayes distance between a pure pattern $\omega = (S,B)$ and one of its normally deformed observed patterns, $\omega' = (S,A)$. If $A = \{a_i | a_i = (s_i,x_i), x_i = (x_{i1},x_{i2},\ldots,x_{iN}), i = 1,2,\ldots,\}$ and $B = \{b_i | b_i = (s_i,w_i), w_i = (w_{i1},w_{i2},\ldots,w_{iN}), i = 1,2,\ldots,n\}$, and assume the following conditions:

(1) Component random variables x_{ij} of x_i are all independent with mean w_{ij}, $j = 1,2,...,N$. An example for this case happens when every x_i is corrupted with random noise with zero mean.

(2) x_{ij} is distributed according to the following normal density function

$$f_{ij}(x_{ij}) = \frac{1}{\sqrt{2\pi}\,\sigma_{ij}} \, EXP\left[-\frac{1}{2}\left(\frac{x_{ij}-w_{ij}}{\sigma_{ij}}\right)^2\right] .$$

(3) Pure pattern ω has the same probability to occur as any other, so that $P(\omega_j)$ is a constant for every pure pattern ω_j.

Then we get the Bayes distance from ω' to ω as

$$\mathcal{B}(\omega',\omega) = -\ln\lambda$$

$$= -\sum_{i=1}^{n}[\ln p(s_i|s_i) + \ln q(x_i|s_i,s_i)] - \ln P(\omega)$$

$$= \sum_{i=1}^{n}\left(\sum_{j=1}^{N}\ln f_{ij}(x_{ij})\right) - \ln P(\omega)$$

$$= K + \sum_{i=1}^{n}\sum_{j=1}^{N}\left[\frac{1}{2}\left(\frac{x_{ij}-w_{ij}}{\sigma_{ij}}\right)^2 + \ln\sigma_{ij}\right] ,$$

where K is a constant, and as far as discrimination is concerned, we can define the normalized square-error distance as

$$B_1(\omega',\omega) = \sum_{i=1}^{n} \sum_{j=1}^{N} [(\frac{x_{ij} - w_{ij}}{\sigma_{ij}})^2 + 2\ln \sigma_{ij}] ,$$

and the (unnormalized) square-error distance as

$$B_2(\omega',\omega) = \sum_{i=1}^{n} \sum_{j=1}^{N} (x_{ij} - w_{ij})^2$$

which is valid under a further assumption that all $\sigma_{ij} = 1$. A SPECP using the normalized or unnormalized least-square-error (LSE) distance criterion is called a normalized or unnormalized LSE SPECP. These two kinds of LSE SPECP for tree languages have been used by Tsai and Fu [5] for the segmentation and recognition of textures corrupted with random noise, and the results show their applicability with the normalized LSE SPECP better than its unnormalized version.

4. Bayes Error-Correcting Recognition System - A Hybrid Pattern Classifier

Given m pattern classes $C_1, C_2, \ldots, C_m$ of pure images and their pattern grammars $G_1, G_2, \ldots, G_m$, after a given input observed pattern ω is parsed by all the Bayes SPECP of the grammars, we get a set of minimum Bayes distances $B(\omega, C_1)$, $B(\omega, C_2), \ldots, B(\omega, C_m)$. Actually, these distances are just the negative logarithms of the conditional probabilities or densities of ω given that $\omega \in C_i$, or

$$p(\omega | C_i) = \mathrm{EXP}[- B(\omega, C_i)] \ ,$$

$i = 1,2,\ldots,m$. Our classification problem is to assign ω to one of these m classes, which has a highest possibility to accept ω as its observed pattern.

Again, we can apply the Bayes decision rule to get

$$P(C_\ell | \omega) = \max_{i=1,2,\ldots,m} P(C_i | \omega) \quad \text{decide} \quad \omega \rightarrow C_\ell \ ,$$

or

$$P(\omega | C_\ell) P(C_\ell) = \max_{i=1,2,\ldots,m} p(\omega | C_i) P(C_i) \quad \text{decide} \quad \omega \rightarrow C_\ell \ ,$$

where $P(C_i)$ is the a priori probability for pattern class C_i, $i = 1,2,\ldots,m$. We call this <u>interclass</u> Bayes classifier together with the <u>intraclass</u> Bayes SPECP a <u>Bayes error-correcting recognition system</u>, compared to the <u>maximum-likelihood classification system</u> set up originally by Fung and Fu [3]. Such a Bayes error-correcting recognition system essentially has also been proposed by Lu and Fu [20] and Fung and Fu [18], but, as mentioned in Section 3.5, the error-correcting capability for substitution errors of their system can only take care of syntactic local deformations. The proposed system here can be considered as a generalization of theirs. Note that in the proposed system, the Bayes decision rule has been used twice for recognition of observed pattern primitives and for classification of the entire observed pattern, and SPECP are used to perform the stochastic syntax parsings of input pattern structural representations. So the recognition system can be regarded as a hybrid pattern classifier because advantages of both syntactic and statistical pattern recognition techniques have been utilized.

Computationally, this system requires more computer time in computing the Bayes distances during parsing if both syntactic or semantic local deformations are involved, but it saves some computer time by avoiding thresholding continuous semantic information existing in the primitives.

Compared with the syntactic recognition approach using stochastic grammars only [7,15], the proposed deformational scheme can be regarded as a special type of stochastic transformational grammars which is expected to handle complex noisy input patterns where simple stochastic grammars may not be adequate to apply [3].

4.1 An Illustrative Example

A complete example for string languages is given in this section to illustrate the applicability of the proposed Bayes error-correcting recognition system and its superiority to other error-correcting systems which handle continuous semantic information by thresholding it into finite discrete cases.

Assume that we have two pure pattern classes. One pattern class C_1 consists of two equilateral triangles ω_{11}, ω_{12}, as shown in Fig. 1(a), and the other class C_2 consists of two other different equilateral triangles ω_{21}, ω_{22} as shown in Fig. 1(b). The primitives used which are fixed-length line segments are shown in Fig. 1(c).

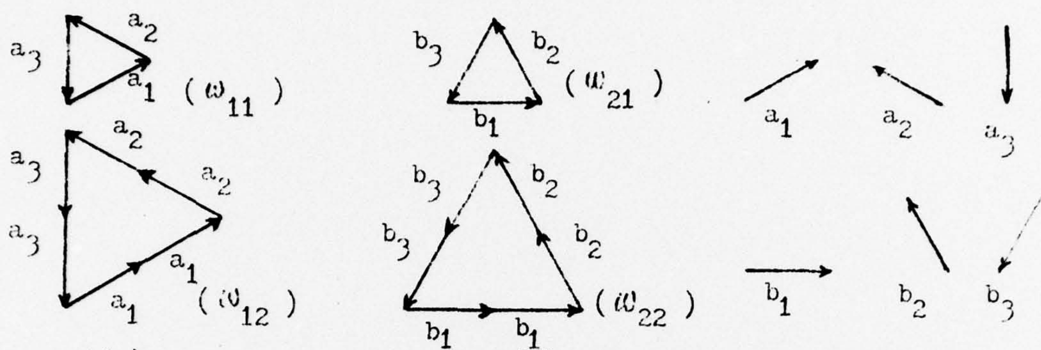

Fig.1(a) Fig.1(b) Fig.1(c)

Also assume the following probability values: $P(C_1) = 0.5$, $P(C_2) = 0.5$, $P(\omega_{11}|C_1) = 0.60$, $P(\omega_{12}|C_1) = 0.40$, $P(\omega_{21}|C_2) = 0.80$, $P(\omega_{22}|C_2) = 0.20$. Two stochastic pattern grammars G_1, G_2, consistent with these probabilities for C_1, C_2, respectively, are as following:

$G_1 = (V_{N1}, V_{T1}, P_1, S_1)$

$V_{N1} = \{A, B, C, D, A_1, B_1, C_1, D_1\}$

$V_{T1} = \{a_1, a_2, a_3\}$

P_1 :

$S_1 \xrightarrow{0.6} AD$ (1)

$S_1 \xrightarrow{0.4} A_1D_1$ (2)

$D \xrightarrow{1.0} BC$ (3)

$A_1 \xrightarrow{1.0} AA$ (4)

$D_1 \xrightarrow{1.0} B_1C_1$ (5)

$B_1 \xrightarrow{1.0} BB$ (6)

$C_1 \xrightarrow{1.0} CC$ (7)

$A \xrightarrow{1.0} a_1$ (8)

$B \xrightarrow{1.0} a_2$ (9)

$C \xrightarrow{1.0} a_3$ (10)

and

$$G_2 = (V_{N2}, V_{T2}, P_2, S_2)$$

$$V_{N2} = \{A, B, C, D, A_1, B_1, C_1, D_1\}$$

$$V_{T2} = \{b_1, b_2, b_3\}$$

P_2 :

$$S_2 \xrightarrow{0.8} AD \qquad (1)$$

$$S_2 \xrightarrow{0.2} A_1D_1 \qquad (2)$$

$$D \xrightarrow{1.0} BC \qquad (3)$$

$$A_1 \xrightarrow{1.0} AA \qquad (4)$$

$$D_1 \xrightarrow{1.0} B_1C_1 \qquad (5)$$

$$B_1 \xrightarrow{1.0} BB \qquad (6)$$

$$C_1 \xrightarrow{1.0} CC \qquad (7)$$

$$A \xrightarrow{1.0} b_1 \qquad (8)$$

$$B \xrightarrow{1.0} b_2 \qquad (9)$$

$$C \xrightarrow{1.0} b_3 \qquad (10).$$

To use the Bayes SPECP of Algorithm 1 for illustrative purpose, the above two grammars are inferred in their context-free forms, although simpler finite-state grammars can certainly be used. They are also in Chomsky normal form.

Now assume that each pattern ω_{ij} $(i = 1,2,\ j = 1,2)$ is subject to both syntactic and semantic local deformations such that each line segment in ω_{ij} is deformed independently. The semantic local deformation is induced only on the direction of each line segment. And each line segment can be syntactically deformed into a curve segment with a fixed curvature and a fixed length but with a variable direction. So we can use the 2-tuple (L,θ) and (C,θ) to characterize the pure primitives - line segments, and the deformed primitives - curve segments, respectively, where L and C are syntactic symbols, and θ denotes the one-dimensional semantic vector --- the direction of the primitives with respect to x-axis. So we have all the 2-tuples for the pure primitives shown in Fig. 1(c) as

$$a_1 = (L,30^o) \qquad b_1 = (L,0^o)$$

$$a_2 = (L,150^o) \qquad b_2 = (L,120^o)$$

$$a_3 = (L,270^o) \qquad b_3 = (L,240^o) \ .$$

And we assume that each a_i $(i = 1,2,3)$ can be deformed syntactically into a curve segment with probability 0.1, and that each b_i $(i = 1,2,3)$ can be deformed syntactically into a curve segment with probability 0.13. Furthermore, each line or curve segment is semantically deformed on its direction θ approximately with a normal distribution as shown in the following data (for notation, see Sec. 2.5):

$$D_{a_i} = \{a_{i1} = a_i = (L,\theta_{a_i}) \ , \ a_{i2} = (C,\theta_{a_i})\}$$

$$\text{where} \quad \theta_{a_i} = 30^o + (i-1) \cdot 120^o$$

$$\text{with} \quad p(a_{i1}|a_i) = 0.9 \ , \ p(a_{i2}|a_i) = 0.1$$

$$i = 1,2,3.$$

$$D_{b_i} = \{b_{i1} = b_i = (L,\theta_{b_i}) \ , \ b_{i2} = (C,\theta_{b_i})\}$$

$$\text{where} \quad \theta_{b_i} = (i-1) \cdot 120^o$$

$$\text{with} \quad p(b_{i1}|b_i) = 0.87 \ , \ p(b_{i1}|b_i) = 0.13$$

$$i = 1,2,3.$$

$$D_{a_{ij}} = \{a_{ijk} | a_{ijk} = (S_j, \theta_k) \ , \ |\theta_k - \theta_{a_i}| \leq 40^{o\dagger}\}$$

where

$$i = 1,2,3, \quad j = 1,2,$$

$$S_j = L \quad \text{when} \quad j = 1$$

$$= C \quad \text{when} \quad j = 2 \ ,$$

and

$$q(a_{ijk} | a_{ij}, a_i) = \frac{1}{\sqrt{2\pi}\ \sigma_a} \text{EXP}[- \frac{1}{2}\left[\frac{\theta_k - \theta_{a_i}}{\sigma_a}\right]^2]$$

with

$$\sigma_a = 8^o \ , \ \theta_{a_i} = 30^o + (i-1) \cdot 120^o \ .$$

$$D_{b_{ij}} = \{b_{ijk} | b_{ijk} = (S_j, \theta_k) \ , \ |\theta_k - \theta_{b_i}| \leq 40^{o\dagger}\}$$

where $i = 1,2,3 \ , \ j = 1,2,$

$$S_j = L \quad \text{when} \quad j = 1$$

$$= C \quad \text{when} \quad j = 2 \ ,$$

and

$$q(b_{ijk} | b_{ij}, b_i) = \frac{1}{\sqrt{2\pi}\ \sigma_b} \text{EXP}[- \frac{1}{2}\left[\frac{\theta_k - \theta_{b_i}}{\sigma_b}\right]^2]$$

†Mathematically, there is no limitation on the value of θ_k, but for computational convenience, let's assume so.

and

$$\sigma_b = 10^\circ \quad , \quad \theta_{b_i} = (i-1) \cdot 120^\circ \ .$$

The 6 semi-pure primitives, i.e., the 6 curve segments corresponding to a_{12}, a_{22}, a_{32}, and b_{12}, b_{22}, b_{32} are shown in Fig. 2(a). Two possible observed patterns deformed from ω_1, ω_2 are shown in Fig. 2(b) and Fig. 2(c), respectively.

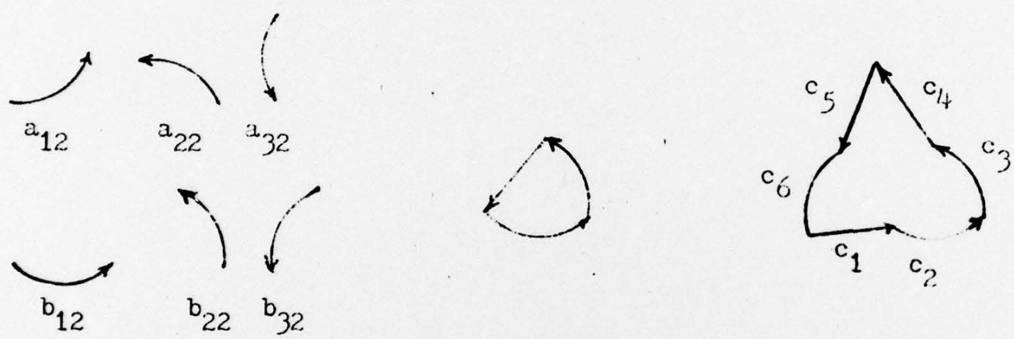

Fig.2(a) Fig.2(b) Fig.2(c)

Now suppose we want to classify the deformed pattern ω' shown in Fig. 2(c) with the following string representation:

$$\omega' = c_1 c_2 c_3 c_4 c_5 c_6$$

where

$$c_1 = (L, 15^\circ), \qquad c_4 = (L, 135^\circ) \ ,$$

$$c_2 = (C, 15^\circ), \qquad c_5 = (L, 255^\circ) \ ,$$

$$c_3 = (C, 135^\circ), \qquad c_6 = (C, 255^\circ) \ .$$

At first, we apply the Bayes SPECP for grammar G_1 and G_2 to ω' respectively, by using the algorithm proposed in Sec. 3.3. When finished, we get the following two parse tables T_1, T_2 for G_1 and G_2, respectively. Since S_1 is in t_{16} of T_1, and S_2 in t_{16} of T_2, ω' is accepted by classes C_1 and C_2 with minimum Bayes distances $d_1 = 36.68$ and $d_2 = 34.19$, respectively.

$(S_1,36.68,2)$					
ϕ	ϕ				
ϕ	ϕ	$(D_1,23.84,5)$			
ϕ	ϕ	ϕ	ϕ		
$(A_1,11.92,4)$	ϕ	$(B_1, 11.92,6)$	ϕ	$(C_1,11.92,7)$	
(A,4.86,8)	(A,7.06,8)	(B,7.06,9)	(C,4.86,9)	(C,4.86,10)	(C,7.06,10)

(Parse Table T_1)

$(S_2,34.19,2)$					
ϕ	ϕ				
ϕ	ϕ	$(D_1,21.72,5)$			
ϕ	ϕ	ϕ	ϕ		
$(A_1,10.86,4)$	ϕ	$(B_1,10.86,6)$	ϕ	$(C_1,10.86.7)$	
(A,4.48,8)	(A,6.38,8)	(B,6.38,9)	(B,4.48,9)	(C,4.48,10)	(C,6.38,10)

(Parse Table T_2)

Next, we apply the interclass Bayes decision rule to get

$$P(C_1|\omega') = p(\omega'|C_1)P(C_1)$$
$$= EXP(-36.68) \cdot 0.5$$
$$= 5.88 \times 10^{-17}$$

$$P(C_2|\omega') = EXP(-34.19) \cdot 0.5$$
$$= 70.87 \times 10^{-17}$$

So we decide that ω' belongs to C_2. This completes our illustrative example for the proposed Bayes error-correcting recognition system.

In the following, we threshold the continuous θ values into intervals as is usually done by other error-correcting schemes, and show how contrary decision can be made for the previous input pattern ω'. Since the proposed Bayes recognition system always gives optimum decision in the Bayes sense, we thus have shown its better performance than other systems using thresholding approaches on continuous sematic information.

If we threshold θ values starting from 0^{o}† in steps of 20^{o} for class C_1, and from 30^{o}† in steps of 20^{o} for C_2, then $D_{a_{ij}}$ and $D_{b_{ij}}$ can be changed to the following:

$$D_{a_{ij}} = \{a_{ijk} | K = 1,2,3,4,\ a_{ijk} = (S_j, \theta_K),\ (K-2)\cdot 20^{o} \le \theta_K - \theta_{a_i} \le (K-1)\cdot 20^{o}\}$$

with <u>discrete</u> probabilities

$$q(a_{ijk}|a_{ij},a_i) = \begin{cases} 0.01, & K = 1,4 \\ 0.49, & K = 2,3, \end{cases}$$

†Starting from different points to threshold is just for convenience, because the directions for a_1, b_1 are 0^{o} and 30^{o}.

$$D_{b_{ij}} = \{b_{ijk} | K = 1,2,3,4,\ b_{ijk} = (S_j, \theta_K),\ (K-2)\cdot 20^{o} \leq \theta_K - \theta_{b_i} \leq (K-1)\cdot 20^{o}\}$$

with discrete probabilities

$$q(b_{ijk}|b_{ij}, b_i) = \begin{matrix} 0.02, & K = 1,4 \\ 0.48, & K = 2,3, \end{matrix}$$

with S_j the same as defined previously. And by convention, only the following probability values are used in parsing [3]:

$$r(a_{ijk}|a_i) = q(a_{ijk}|a_i, a_j)\cdot p(a_{ij}|a_i) = \begin{cases} 0.009, & j = 1,\ K = 1,4 \\ 0.441, & j = 1,\ K = 2,3 \\ 0.001, & j = 2,\ K = 1,4 \\ 0.049, & j = 2,\ K = 2,3 \end{cases}$$

$$r(b_{ijk}|b_i) = q(b_{ijk}|b_i, b_j)\cdot p(b_{ij}|b_i) = \begin{cases} 0.0174, & j = 1,\ K = 1,4 \\ 0.4176, & j = 1,\ K = 2,3 \\ 0.0026, & j = 2,\ K = 1,4 \\ 0.0624, & j = 2,\ K = 2,3 \end{cases}$$

$i = 1,2,3$. The previous data shows that each a_i or b_i can be deformed into 8 different observed primitives with different probabilities, in which four are line segments and the other four are curve segments.

Now again use the Bayes SPECP proposed in Sec. 3.3 for G_1, G_2 to parse ω', respectively. Note that after thresholding the θ values in ω' and transforming into string representations, we get

$$\omega' = a_{113}a_{123}a_{223}a_{213}a_{313}a_{323}$$

for class C_1, or

$$\omega' = b_{112}b_{122}b_{222}b_{212}b_{312}b_{322}$$

for class C_2. Also note that the term $[\ln p(s_i|t_j) + \ln q(x_i|t_j, s_i)]$ in Algorithm 1 should be replaced by $\ln \hat{q}(c_i|a_j)$ before the algorithm is applied to our discrete case here, where $c_i = a_{ijk}$ or b_{ijk} now.

(S_1,12.44,2)					
ϕ	ϕ				
ϕ	ϕ	(D_1,7.68.5)			
ϕ	ϕ	ϕ	ϕ		
(A_1,3.84,4)	ϕ	(B_1,3.84,6)	ϕ	(C_1,3.84,7)	
(A,0.82,8)	(A,3.02,8)	(B,3.02,9)	(B,0.82,9)	(C,0.82,10)	(C,3.02,10)

(Parse Table T_1)

(S_2,12.53,2)					
ϕ	ϕ				
ϕ	ϕ	(D_1,7.28,5)			
ϕ	ϕ	ϕ	ϕ		
(A_1,3.64,4)	ϕ	(B_1,3.64,6)	ϕ	(C_1,3.64,7)	
(A,0.87,8)	(A,2.77,8)	(B,2.77,9)	(B,0.87.9)	(C,0.87.10)	(C,2.77,10)

(Parse Table T_2)

From the above tables, we get

$$P(C_1|\omega') = EXP(-12.44) \cdot 0.5$$

$$= 1.98 \times 10^{-6}$$

$$P(C_2|\omega') = EXP(-12.53) \cdot 0.5$$

$$= 1.81 \times 10^{-6}$$

So we decide ω' belongs to C_1!

A careful study reveals that such contrary conclusion to the previous Bayesian decision $\omega' \rightarrow C_2$ is due to the rough thresholding used. Using smaller intervals in thresholding will improve the result, but never be better than our proposed system which has minimum probability of errors for recognition of primitives due to the use of the Bayes rule in the error-correcting parser.

5. Concluding Remarks

Bayes error-correcting recognition systems using Bayes error-correcting parsers and Bayes interclass decision rule have been proposed both by Fung and Fu [18] and by Lu and Fu [20]. The proposed system in this report can be considered, from the viewpoint of local deformations, as a generalization of theirs in the aspect of semantic information, which is more relevant for practical pattern classifications where both structural and numerical informations are available for primitive discrimination, as emphasized by several investigators [13,19,6]. Further investigations should be directed to include error-correcting capability for structural deformations under the formalism of the proposed deformational model and thus set up a more complete error-correcting recognition system for more practical applications.

Bibliography

[1] Aho, A. V. and Peterson, T. G., "A Minimum Distanct Error-Correcting Parser for Context-Free Languages," SIAM J. Comput., Vol. 1, pp. 305-312, Dec. 1972.

[2] Thompson, R. A., "Language Correction using Probabilistic Grammars," IEEE Trans. on Comput., Vol. C-25, No. 3, Mar. 1976.

[3] Fung, L. W. and Fu, K. S., "Stochastic Syntactic Decoding for Pattern Classification," IEEE Trans. on Comput., Vol. C-24, No. 6, July 1975.

[4] Lu, S. Y. and Fu, K. S., "Structure-Preserved Error-Correcting Tree automata for Syntactic Pattern Recognition," IEEE Conf. on Decision and Control, Dec. 1-3, Clearwater Beach, FL, 1976.

[5] Tsai, W. H. and Fu, K. S., "Image Segmentation and Recognition by Texture Discrimination: A Syntactic Approach," to be presented at the 4th International Joint Conference on Pattern Recog., KYOTO, JAPAN, Nov. 7-10, 1978.

[6] Chen, C. H., "On Statistical and Structural Feature Extraction," Joint Workshop on Pattern Recog. and Art. Intellig., Hyannis, MA, June 1976.

[7] Fu, K. S., Syntactic Methods in Pattern Recognition, New York: Academic Press, 1974.

[8] Narasimhan, R., "On the Description, Generation, and Recognition of Classes of Pictures," in Automatic Interpretation and Classification of Images (A. Grasselli, ed.), Academic Press, New York, 1969.

[9] You, K. C. and Fu, K. S., "Syntactic Shape Recognition using Attributed Grammars," 8th Annual Automatic Imagery Pattern Recognition, April 3-4, 1978, Gaithersburg, Maryland.

[10] Lu, S. Y. and Fu, K. S., "A Syntactic Approach to Texture Analysis," Compt. Graphics and Image Processing, June 1978.

[11] Lu, S. Y. and Fu, K. S., "Error-Correcting Tree Automata for Syntactic Pattern Recognition," Proc. 1978 IEEE Computer Society Conf. on Pattern Recognition and Image Processing, Troy, N.Y.

[12] Bhargava, B. K. and Fu, K. S., "Transformation and Inference of Grammars for Syntactic Pattern Recognition," Proc. 1974 IEEE Int'l. Conf. on Cybernetics and Society, Oct., Dallas, Texas.

[13] Grenander, U., "A Unified Approach to Pattern Analysis," in Advances in Computers, Vol. 10, New York: Academic, 1970.

[14] Kovalevsky, V. A., "Sequential Optimization in Pattern Recognition and Pattern Description," in Proc. Int. Fed. Info. Process. Congr., Amsterdam, the Netherlands, 1968.

[15] Lee, H. C. and Fu, K. S., "A Syntactic Pattern Recognition System with Learning Capability," Int. Symp. Comput. Inform. Sci., 4th Miami Beach, Florida, Dec. 1972.

[16] Aho, A. V. and Ullman, J. D., The Theory of Parsing, Translation, and Compiling, Vol. 1, Englewood Cliffs, N.J.: Prentice-Hall, 1972.

[17] Fu, K. S. and Bhargava, B. K., "Tree Systems for Syntactic Pattern Recognition," IEEE Trans. on Computers, Vol. C-22, No. 12, Dec. 1973.

[18] Fung, L. W. and Fu, K. S., "Syntactic Decoding for Computer Communication and Pattern Recognition," Purdue University, TR-EE 74-47, Dec. 1974.

[19] Kanal, L. and Chandrasekaran, B., "On Linguistic Statistical, and Mixed Models for Pattern Recognition," in Frontier of Pattern Recognition, S. Watanabe, ed., Academic Press, Inc., New York, N.Y., 1972.

[20] Lu, S. Y. and Fu, K. S., "Stochastic Error-Correcting Syntax Analysis for Recognition of Noisy Patterns," IEEE Trans. on Computers, Vol. C-26, No. 12, Dec. 1977.

[21] Shaw, A. C., "A Formal Picture Description Scheme as a Basis for Picture Processing Systems," Inform. and Control, Vol. 14, 9-52 (1969).

[22] Feder, J., "Plex Language," Information Sci., Vol. 3, 225-241 (1971).

BIBLIOGRAPHIC DATA SHEET	1. Report No. AFOSR-TR- 78-1297	2.	3. Recipient's Accession No.

4. Title and Subtitle

A PATTERN DEFORMATIONAL MODEL AND BAYES ERROR-CORRECTING RECOGNITION SYSTEM

5. Report Date
May 1978

6.

7. Author(s)
W. H. Tsai and K. S. Fu

8. Performing Organization Rept. No.

9. Performing Organization Name and Address

Purdue University
School of Electrical Engineering
West Lafayette, IN 47907

10. Project/Task/Work Unit No.

11. Contract/Grant No.
MDA-903-77-G-1 and
7873-53-12855

12. Sponsoring Organization Name and Address

Defense Advanced Res. Proj. Agency
1400 North Wilson Blvd.
Arlington, VA 22209

AFOSR/NM
Bldg. 410
Bolling AFB
Washington, D.C. 20332

13. Type of Report & Period Covered
Technical Report

14.

15. Supplementary Notes

16. Abstracts

Various types of pattern deformations are investigated from the syntactic point of view and categorized into two major types: local deformations and structural deformations. Random noise, distortion variations, and substitutions, of pattern primitives belong to the former; syntactic errors due to pattern structural changes, such as primitive deletions and insertions, belong to the latter. Every observed pattern can be regarded as transformed from a pure pattern through these two types of deformations. An error-correcting parsing scheme for local deformations optimum in the Bayes sense is proposed. A corresponding recognition rule is then described, which can be regarded as a hybrid classifier because it has utilized advantages of both syntactic and statistical approaches to pattern recognition. When this scheme is applied to string and tree languages without structural deformations, it is shown that various known structure-preserved error-correcting parsing schemes could be considered as special cases of this general scheme. Two structure-preserved error-correcting parsers, one for string languages, the other for tree languages, are also presented. Finally, further researchers concerning error-correcting parsings for structural deformations and a complete error-correcting system for both kinds of deformations are suggested.

17b. Identifiers/Open-Ended Terms

17c. COSATI Field/Group

18. Availability Statement	19. Security Class (This Report) UNCLASSIFIED	21. No. of Pages 43
	20. Security Class (This Page) UNCLASSIFIED	22. Price